REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
02-08-2007	Final	10-02-2000 / 30-09-2006

4. TITLE AND SUBTITLE

a. The fine grain level of tutorial discourse
b. A computational model of expert tutoring

5a. CONTRACT NUMBER

5b. GRANT NUMBER
N00014-00-1-0640

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Barbara Di Eugenio

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

The Board of Trustees
University of Illinois at Chicago
Grants and Contracts (M/C 551)
809 S. Marshfield Avenue, Chicago IL 60612

8. PERFORMING ORGANIZATION REPORT NUMBER
N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Office of Naval Research
Ballston Center Tower One
800 North Quincy Street
Arlington VA 22217-5660

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

We study tutorial dialogue with two aims: understanding what promotes learning in one on one tutoring; developing language interfaces to Intelligent Tutoring Systems (ITSs). We worked in three different domains. Our work comprises: linguistic analysis, data mining, computational modeling (e.g., discourse planning), implementation, and empirical evaluation with human subjects. Our results show that interfaces developed on the basis of the tutorial dialogue analysis engender significantly more learning than other types of interfaces.

15. SUBJECT TERMS

Intelligent Tutoring Systems; Interfaces; Tutorial dialogue

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Barbara Di Eugenio
U	U	U	UU	14	19b. TELEPHONE NUMBER *(Include area code)* 312 996 7566

Reset

Final Report for Award ONR No. N00014-00-1-0640

Barbara Di Eugenio

August 2, 2007

Abstract

Award ONR No. N00014-00-1-0640 funded Dr. Barbara Di Eugenio for 6 years, from October 2000 to September 2006, first as a single PI (2000-2003), then with Dr. Stellan Ohlsson as co-PI (2003-2006). The topic of the award was Natural Language dialogue for Intelligent Tutoring Systems; because of the length of the award, three distinct projects were covered by this award. They will be presented in turn in this report.

1 Introduction

Intelligent Tutoring Systems (ITSs) are computer systems that help students learn a certain subject. The ideal ITS would be indistinguishable from a good human tutor, thus affording students individualized help on demand. Many researchers in the field believe that one integral component of this ideal ITS will be the ability to engage a student in a conversation, a *tutorial dialogue* to be precise; to do so, the student would communicate with the ITS via his / her own Natural Language (NL) [Evens *et al.*, 1993; Rosé *et al.*, 2003; Litman *et al.*, 2004; Peters *et al.*, 2004; Graesser *et al.*, 2005; Di Eugenio *et al.*, 2005a; Zinn *et al.*, 2005; Di Eugenio *et al.*, 2006; Evens and Michael, 2006; Litman *et al.*, 2006; Pon-Barry *et al.*, 2006].

Given this general framework, our general *scientific and technical objectives* over the six years were articulated as follows:

- to analyze tutorial dialogues in order to uncover what makes them effective, and further, to contribute to the development of theories of learning;
- to devise computational architectures that support tutorial dialogue, in particular the generation of feedback to the student, and that can be easily integrated into ITSs;
- to show that ITSs endowed with language interfaces engender higher learning scores than ITSs without such interfaces.

An orthogonal objective was to experiment with a variety of techniques to support the difficult task of human dialogue annotation.

Our *approach* follows a general methodology that can be decomposed as follows:

1. Collection of human tutorial dialogues
2. Development and evaluation of coding schemes to annotate those dialogues – the information we seek cannot be obtained by simply counting words or phrases
3. Data mining of the annotated corpus by means of statistical and machine learning techniques, so as to extract effective tutoring strategies

4. Development of computational models of tutorial dialogues, e.g. dialogue system architectures, probabilistic rules, based on the information extracted from the corpus

5. Development of an ITS with language interface that implements the model(s) just described

6. Controlled evaluation of the effectiveness of the ITS with user studies; for example, the full ITS may be compared to a paired down version where the capabilities of the language interface have been constrained

While the methodology is well established in its abstract form, in the course of the six years and the three different domains we worked in, we have come to a better understanding of many of the steps mentioned above. We will point out the changes we brought to this general methodology below.

2 The DIAG-NLP project

In the first few years of the award, we focused on adding language capabilities to an existing ITS, or to be more precise, ITS authoring shell. DIAG [Towne, 1997] is a shell to build ITSs based on interactive graphical models that teach students to troubleshoot complex systems such as home heating and circuitry (one application was an ITS to diagnose malfunctions in a radar). DIAG uses very simple templates to assemble the text to present to the student. As a result, its feedback is highly repetitive. Our goal was to assess whether simple, rapidly deployable NL generation techniques would lead to measurable improvements in the student's learning. We focused on aggregation, namely, on how to group information in sentences. We developed two different feedback generation engines, *DIAG-NLP1* and *DIAG-NLP2*, that we systematically evaluated in a three way comparison that included the original system (*DIAG-orig*) as well. In *DIAG-NLP1* we focused on aggregating information at the syntactic level, and by exploiting the hierarchical structure of the system; in *DIAG-NLP2* we went one step further and aggregated information at a functional level. Fig. 1 shows the replies provided by each system to the same query, a request of information regarding the indicator *visual combustion check - indicators* are those system components the student can test.

The kind of aggregation we modeled in *DIAG-NLP2* was empirically grounded in human data. We collected 23 tutoring interactions between a student using the DIAG tutor on home heating and one of two human tutors. This amounts to 272 tutor turns. We developed a coding scheme [Glass *et al.*, 2002] and annotated the data. We found that tutors provide explicit problem solving directions in 73% of the replies, and evaluate the student's action in 45% of the replies. As expected, they *exclude* much of the information (63% to be precise) that DIAG would provide, and specifically, always exclude any mention of parts that are not as likely to cause a certain problem, e.g. the *ignitor assembly* in Fig. 1. Tutors do perform a fair amount of aggregation, as measured in terms of the number of parts and indicators labelled as *summary*. Further, they use functional, not syntactic or structural, aggregation of parts. E.g., the oil nozzle, supply valve, pump, filter, etc., are described as *the path of the oil flow*.

2.1 Results

Our empirical evaluation is a between-subject study with three groups: the first interacts with *DIAG-orig*, the second with *DIAG-NLP1*, the third with *DIAG-NLP2*. The 75 subjects (25 per group) were all science or engineering majors affiliated with our university. Each subject read some short material about the domain (diagnosing a home heating system), went through one trial problem, then continued through the curriculum on his/her own. The curriculum consisted of three problems of increasing difficulty. As there was no time limit, every student solved every problem. Reading materials and curriculum were identical in the three conditions.

The visual combustion check is igniting which is abnormal (normal is combusting).
Oil Nozzle always produces this abnormality when it fails.
Oil Supply Valve always produces this abnormality when it fails.
Oil pump always produces this abnormality when it fails.
Oil Filter always produces this abnormality when it fails.
System Control Module sometimes produces this abnormality when it fails.
Ignitor Assembly never produces this abnormality when it fails.
Burner Motor always produces this abnormality when it fails.

The visual combustion check indicator is igniting.
This is abnormal.
Normal is combusting.

Within the furnace system,
  this is sometimes caused if
    the System Control Module has failed.

Within the Oil Burner
  this is never caused if
    the Ignitor Assembly has failed.
  In contrast, this is always caused if
    the Burner Motor, Oil Filter, Oil Pump, Oil Supply Valve, or Oil Nozzle has failed.

The combustion is abnormal.
In the oil burner, check the units along the path of the oil and the burner motor.

Figure 1: Answers by *DIAG-orig*, *DIAG-NLP1* and *DIAG-NLP2* to same query

At the end of the experiment, each subject was administered a post-test, a test of whether subjects remember their actions, and a usability questionnaire. We found that subjects who used *DIAG-NLP2* had significantly higher scores on the post-test, and were significantly more correct in remembering what they did. Performance on individual questions in the post-test is illustrated in Fig. 2. Scores in *DIAG-NLP2* are always higher, significantly so on questions 2 and 3 ($F = 8.481, p = 0.000$, and $F = 7.909, p = 0.001$), and marginally so on question 1 ($F = 2.774, p = 0.069$).[1] As regards usability, subjects prefer the NL enhanced systems to *DIAG-orig*, however results are mixed as regards which of the two they actually prefer.

In conclusion, this work was among the first to show that a NL interaction improves learning. This research led to two prestigious publications [Di Eugenio *et al.*, 2005a; Di Eugenio *et al.*, 2005b], at ACL05 and AIED05 respectively, including a nomination for *Best Paper Award* at AIED05; a journal paper is currently under revision and is likely to be published by the International Journal of AI in Education.

In addition, the aggregation techniques developed for DIAG-NLP were applied in a completely different domain, that of generating driving directions [Di Eugenio and Trolio, 2005].

3 Towards a theory of *expert* tutoring

Via DIAG-NLP, the PI had shown that NL did make a difference, and more specifically, that an NL interface modeled on the behavior of human tutors was effective. The reader should note this is not a foregone conclusion by any means. First, the field did not precisely know, and still doesn't, why human tutoring is effective, hence, a priori we do not know whether language positively impacts learning; second, if it does, the issue is which specific features of the language interaction are responsible for learning. *DIAG-NLP2*

[1] Significant differences were determined by an analysis of variance performed via ANOVA, followed by Tukey post-hoc tests if necessary.

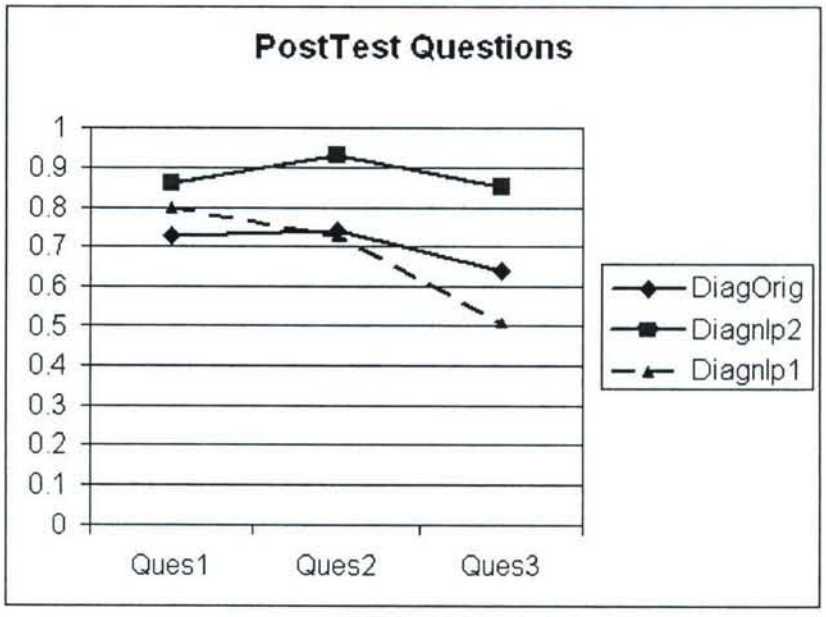

Figure 2: Scores on problems

had shown that more directive and more abstract feedback was more effective than other types of feedback, however, this result in a sense begged the question: why was that the case? namely, how did the features of the language feedback provided by the system promote learning?

At that point, Dr. Di Eugenio and Dr. Ohlsson, a noted expert on learning, together embarked on a more ambitious plan, to investigate how one-on-one tutoring as embodied in tutorial dialogue can foster learning.

One weakness of the DIAG-NLP work was that the human data was collected with two tutors who were not experienced; although *DIAG-NLP2* was effective, we wondered whether it might have been even more effective if it had been based on the data obtained from more experienced tutors. Thus, we turned to exploring the differences between non expert and expert tutors. When we started this analysis, and still at the time of this writing, very few contrastive studies have been conducted to compare expert vs non expert tutors (e.g. [Glass *et al.*, 1999]), even if there are studies on either novice tutors (e.g., [Graesser *et al.*, 1995]) or experienced tutors (e.g., [Lepper *et al.*, 1997; Person, 2006]).

We collected data and developed computational models in two distinct domains, which we will describe in what follows. Our work in those two domains has reshaped our thinking of the whole approach, as we will discuss under *Conclusions*.

3.1 The letter pattern domain

This domain concerns extrapolating complex letter patterns[Kotovsky and Simon, 1973], which is a well known task for analyzing human information processing in cognitive science. Given a sequence of letters that follows a particular pattern, the student is asked to find the pattern and create a new sequence from a new starting letter. For example, the pattern of the sequence "ABMCDM" is: "M" as a chunk marker separates the whole sequence into two chunks of letters progressing according to the alphabet. Then with a starting letter "E", to maintain this pattern, the student needs to finish the sequence as "EFMGHM". Only knowledge of the alphabet is required in this domain. During the training session, each student went through a curriculum of 13 problems of increasing complexity. To test performance, each student also needs to solve two post-test problems, each 15 letters long, via a computer interface.

We collected tutoring dialogues with three tutors, one expert, one novice, and one whom we call *lec-*

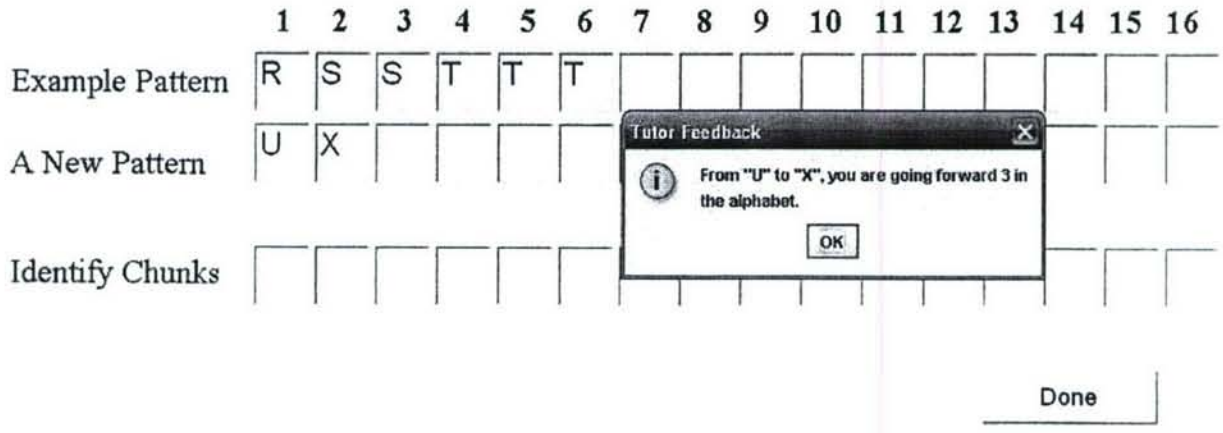

Figure 3: A feedback message in the *model* ITS

turer, because he was experienced in teaching, but not in one-on-one tutoring. Comparison of the student's performance showed that the expert tutor was significantly more effective than the other two tutors.

The dialogues on two specific problems in the curriculum were transcribed and annotated from the videotapes which recorded the tutors' interaction with the subjects. For each tutor, six subjects' dialogues were transcribed; each utterance was annotated with one tutor or student moves.[2] The tutor moves include four high level categories, *reaction, initiative, support, conversation*. The categories *reaction* and *initiative* were further subcategorized into: *answering, evaluating, summarizing*, and *prompting, diagnosing, instructing, demonstrating*, respectively (the annotation scheme was inspired by the literature[Chi *et al.*, 2001; Litman *et al.*, 2006] and adapted to our dialogues).

Classification based on associations (CBA) [Liu *et al.*, 1998], a data mining algorithm, was run on the expert tutor dialogues. The CBA rules were used to drive a dialogue planning module that was the core of the user interface to an ITS in this domain. The more sophisticated ITS that provides language feedback modelled on the expert tutor (we will call it the *model* ITS) was evaluated in a 5 fold comparison, as against 4 different version of a baseline ITS.

Table 1 represents the feedback types of the four baseline versions. Feedback is given for each input letter. Positive and negative verbal feedback are natural language feedback messages which are given out when the student makes a correct action or a wrong action. The positive feedback messages confirm the correct input and explain the relationships which this input is involved in. The negative feedback messages flag the incorrect input and deliver hints.

Table 1: Feedback Types of Four Versions of the baseline ITS

	Version / Feedback	Color	Positive Verbal	Negative Verbal
1	No feedback	No	No	No
2	Color only	Yes	No	No
3	Negative	Yes	No	Yes
4	Positive	Yes	Yes	No

[2]Trere are some rare cases in which an utterance is annotated with more than one move.

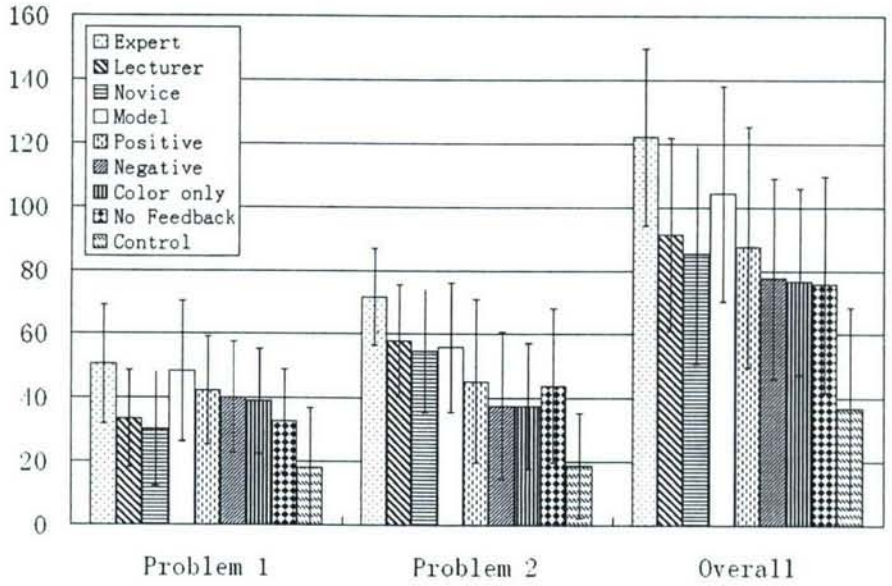

Figure 4: Post-test performance of tutors and five versions of the ITS

3.1.1 Results and significance

From the data analysis point of view, our most interesting finding was that our expert tutor behaved differently from the predictions from the literature. Compared to the lecturer, the expert tutor does less specific prompting and his students explain less. This contradicts the claim that students learn best when they construct knowledge by themselves, and that as a consequence, the tutor should prompt and scaffold students, and leave most of the talking to them [Chi *et al.*, 2001]. This led us to look for other aspects that make the expert tutor more effective. We found that the expert tutor does much more *procedural instructing* (he provides problem solving tips), *demonstrating* (the tutor demonstrates the solution) and *supporting* (the tutor provides emotional support) than the non-expert tutors. Consistently, the novice tutor does much more *declarative instructing* (she provides domain knowledge or facts about the specific problem) [Di Eugenio *et al.*, 2006; Lu, 2006].

Our **main result** [Lu, 2007] is that the ITS modeled on the expert tutor was indeed more effective than any of the other ITSs, and was indistinguishable from the expert human tutor – as summarized in Figure 4.

Comparing post-test performance across the groups, the main findings are (all the statistical results are based on ANOVAs, followed by Tukey post-hoc tests):

- A main effect of ITS ($p \leq 0.05$): The subjects who interacted with the ITSs did significantly better in both the post-test problems than the subjects in the control condition.

- No main effect of simple feedback ($p > 0.05$): The subjects who are trained by the three versions of the ITS with simple feedback (color only, negative, positive) did not have significantly higher post-test scores than the subjects with the "no feedback" version in either of the post-test problems.

- No main effect of simple feedback message ($p > 0.05$): the subjects who are trained by the two versions of the ITS with simple capsulated feedback messages (negative, positive) did not have significantly higher post-test scores than the subjects with the two versions without simple capsulated feedback messages (no feedback, color only) in either of the post-test problems.

- A main effect of natural language feedback message ($p < 0.05$): the subjects who interacted with the *model* version of the ITS did significantly better in the total post-test scores of the two problems than the subjects with any other version of the ITS.

In addition, when comparing human tutors and ITSs, the group of subjects with the expert tutor did significantly better in the post-test than the groups with the four versions of the baseline ITS (no feedback, color only, negative, positive), but did not do significantly better than the group with the *model* version ITS.

A further regression analysis on the factors that affect the post-test score with the *model* ITS showed that beyond the expected factors such as training time and number of errors, the moves by the model ITS that significantly affect post-test score are *evaluating, instructing and demonstrating* – confirming that providing problem solving tips and demonstrating the solution are important moves in both human and artificial tutoring.

This result is very significant because it provides further evidence for the hypothesis that language positively affects learning (not a foregone conclusions, as discussed under the DIAG-NLP project); and starts to precisely pinpoint which specific behaviors are responsible for this effect. The reader may also note an interesting connection to the DIAG-NLP project. The expert tutor is more directive than the other two tutors, and the *model* ITS reflects it (via the fact that *procedural instructing* is correlated with learning). One feature of the more effective feedback from DIAG-NLP was that it was indeed more directive. As we discussed above this result is in contrast with the literature, and now we have two supporting pieces of evidence for it.

3.2 The Computer Science domain

One weakness of the letter pattern study is that the problems to be solved are not problems a student would face in a real school curriculum. Hence, we started working on a real domain, that of introductory Computer Science (CS). This domain was chosen because of Dr. Di Eugenio's interests as an educator, and because, surprisingly enough, educational technology has not been used to support instruction in the fundamentals of Computer Science, i.e., data structures and algorithms.

As a first step, we engaged in an extensive Computer Science tutoring dialogue collection. We collected data from 76 subjects altogether, 28 control and 48 tutored. The 48 tutorial dialogues were videotaped.

Subjects were recruited between majors and non-majors taking introductory CS classes and among graduate students in other engineering departments who would attest to having only a basic knowledge of Computer Science. All subjects, after providing informed consent, are presented with a pre-test. They are allotted 15' to take the pre-test, which they take at their own pace. The pre-test consists of 8 problems, concerning three topics: two problems on linked lists, two on stacks, and four on binary search trees. After the specific activity that each group is engaged in, subjects are given a post-test. The post-test is identical to the pre-test and administered in the same way.

The control subjects were not tutored, but read a handout on the topics of interest. The tutored subjects were divided into two groups according to the tutor they would interact with: a more experienced tutor, a retired Math and Computer Science professor from a small liberal art school, with thirty years of experience in one-on-one tutoring; a less experienced tutor, a senior in Computer Science, with just a few hours under his belt as a volunteer tutor for some introductory classes. Other than the tutor, the tutoring sessions were exactly the same.

3.2.1 Preliminary Analysis

We have preliminary results on the students' learning gains in the different conditions. Specifically, we found

- overall, students learn significantly in each condition
- as concerns the cumulative learning gain across all 8 problems, there are significant differences between each tutored group of subjects and the control group, but not between the two tutored groups.
- probing learning gains on individual problems, the expert tutor always engenders significant learning gains, whereas the less experienced tutor does so in 5 problems out of 8; subjects in the control condition learn significantly more in 4 problems out of 8. However, there are no significant differences between expert and novice tutor as regards average learning gains. There are differences between each tutored group of subjects and the control group for two specific problems.

3.2.2 The core Computer Science ITS

We implemented a first version of the "core" ITS in the Computer Science domain – namely, the ITS backbone itself, without the language interface – based on TDK [Koedinger *et al.*, 2003], in turn based on the model tracing paradigm [Anderson *et al.*, 1995]. We later reimplemented the system within a different theoretical paradigm, that of constraint-based tutors [Ohlsson, 1992; Mitrović and Ohlsson, 1999; Mitrović *et al.*, 2006; Mitrović and Ohlsson, 2006]. The constraint-based approach supports different types of tutoring interventions more easily than model-tracing ITSs, which intervene only when the student makes an error and by providing instruction that helps the student overcome the error.

The current ITS has a fully implemented curriculum as regards linked lists. This curriculum consists of five problems. Students are able to interactively execute atomic operations on linked lists, and visualize the results of the process. The graphical interface is composed by six main areas (please refer to Figure 5): problem area, feedback history, variables space, heap space, operations sequence, command panel.

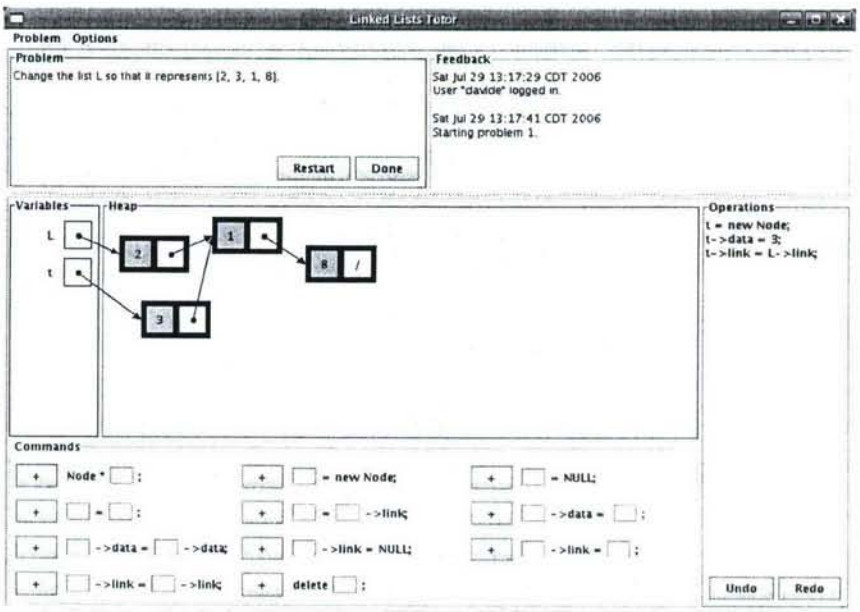

Figure 5: The graphical interface for the Computer Science ITS

When the student thinks the solution is complete, he/she presses the done button. At this time, the system evaluates the solution properties, looking for constraint violations. The system is able to catch some typical mistakes and provide simple feedback in reaction to those mistakes.

Other topics of interest, stacks and binary trees, are being developed. This CS-ITS is being developed with input from faculty from the Naval Academy in Annapolis, MD.

3.2.3 Results and significance

No extensive corpus of tutorial dialogues in Computer Science had been previously collected – in this, we contribute to both furthering Computer Science education, and the Intelligent Tutoring Systems field, in that there is now a different domain in which to verify theories and models.

The ITS we are developing is one of the very few ITSs devoted to CS topics, and basically the only one that teaches data structures and algorithms as opposed to a programming language, or to more advanced topics. Since a full understanding of data structures and algorithms is at the core of Computer Science, we believe our ITS has the potential of deeply affecting CS education. This belief is shared by our collaborator at the Naval Academy in Annapolis, Prof. Christopher Brown, who is helping us shape the system. Our work has thus the potential of positively impacting the training that the Navy requires of its personnel.

Finally, one important result from this initial part of the project was to change our approach, as described under Conclusions. We are continuing data analysis, computational modeling, and system development in the CS domain under our current award from ONR, # N000140710040.

4 Other technical contributions

4.1 Coefficients of intercoder reliability

Speech and text corpora, such as tutorial dialogues, augmented with linguistic annotations have become essential for research. In the realm of discourse and dialogue-related annotation, which we are interested in, linguistic annotation is still mostly a manual effort. Intercoder reliability is an important feature of any annotation effort. It is crucial to establish whether the categories of coding are somewhat objective, or are so subjective that coders cannot agree to a reasonable degree. The standard to compute intercoder reliability in computational linguistics is to use coefficients belonging to the κ or α family of statistics. These coefficients originated in psychology, medicine and content analysis [Scott, 1955; Cohen, 1960; Krippendorff, 1980]. They factor out expected agreement, namely, the probability that the coders agree by chance.

[Carletta, 1996] was the first to draw the attention of the Computational Linguistics community to the issue of intercoder reliability. That paper was extremely influential, and the community rushed to embrace that proposal, but very often without any critical thinking. Dr. Di Eugenio and Dr. Glass authored a widely cited paper [Di Eugenio and Glass, 2004], to highlight the pitfalls of such a blind adoption – Dr. Glass was a postdoctoral fellow supported on this ONR award from 2000 to 2002.

4.2 MUP: An annotation tool

Another important component to support linguistic annotation is the availability of coding tools that facilitate a human coder's task. Under this ONR award, we developed MUP [Glass and Di Eugenio, 2002], a coding tool for standoff markup which is sophisticated enough to allow for a variety of different markings to be applied, but which is also simple enough to use that it does not require a sizable set up effort. In standoff markup [Thompson and McKelvie, 1997] the source text is inviolate and the annotations are kept physically separate, usually in other files. Annotable items in the source text contain labels, while the physically separate annotations refer to these labels. Since annotations are themselves labeled, complex structures of linked annotated constituents pointing to each other are representable. MUP also provides support for computing intercoder reliability, specifically the κ coefficient discussed above.

4.3 Latent Semantic Analysis and Dialogue Act Classification

We extended Latent Semantic Analysis (LSA) [Landauer and Dumais, 1997] with a variety of features to perform dialogue act interpretation and evaluate text coherence – we call the new method FLSA (for

Feature LSA). LSA learns from co-occurrence of words in collections of texts. It builds a semantic space where words and passages are represented as vectors. Their similarity is measured by the cosine of the angle between the two vectors in the semantic space. LSA is based on Single Value Decomposition (SVD), a mathematical technique that causes the semantic space to be arranged so as to reflect the major associative patterns in the data, and ignores the smaller, less important influences.

We used LSA and FLSA for dialogue act classification, namely, to understand which dialogue move (such as *evaluating* or *instructing*), is performed by each utterance. We tested basic LSA on three different corpora, including the DIAG-NLP corpus, for dialogue act classification. Plain LSA reduces error rates between 33% and 52%. Feature LSA reduces error rate between 60% and 67%. FLSA uses features such as the previous dialogue act, who the speaker is, etc.

This work resulted in another prestigious conference publication, [Serafin and Di Eugenio, 2004]. A journal paper is under revision.

5 Graduate students and Postdoctoral fellows

Over the six years, this award directly or indirectly supported 10 graduate students in Computer Science, of which six are women, and 3 graduate students in Psychology, of which two are women. In turn, this resulted in four MS projects, three MS theses, two PhD theses in Computer Science, and two PhD theses in Psychology. In addition, Dr. Michael Glass was supported on this grant as a postdoctoral fellow from Summer 2000 to Summer 2002. He is currently an Assistant Professor in the Math and Computer Science department, Valparaiso University.

6 Authored papers

This award has resulted in a number of published papers, specifically 6 journal papers [Di Eugenio, 2001; Di Eugenio and Glass, 2004; Nokes and Ohlsson, 2005; Mitrović *et al.*, 2006; Mitrović and Ohlsson, 2006; Poesio *et al.*, 2006]; 5 major peer-reviewed conference papers [Di Eugenio *et al.*, 2002; Serafin and Di Eugenio, 2004; Di Eugenio *et al.*, 2005a; Di Eugenio *et al.*, 2005b; Zakharov *et al.*, 2005], 13 other peer-reviewed conference papers, and 2 book chapters – please note that some conferences in Computer Science, such as ACL and AIED, are extremely selective and accept only between 20 and 30% of the submissions; also note that the paper [Di Eugenio *et al.*, 2005b] was named for *Best Paper Award*. In addition, a book [Ohlsson, 2007] and five more journal papers under revision or in preparation, are all based in full or in part on the work this award supported.

7 Conclusions: Towards a theory of *effective* tutoring

Our efforts over the 6 years of this award, and the efforts by many others in the community over an even longer period of time [Chi *et al.*, 2001; Evens and Michael, 2006; Fox, 1993; Graesser *et al.*, 1995; Moore *et al.*, 1996; VanLehn *et al.*, 2003], still have not resulted in agreement on a repertoire of effective tutoring strategies. We have come to believe this is because everybody, including ourselves, has been equating effectiveness with frequency: namely, what tutors, in particular expert tutors, do most often is what is deemed to be effective. However, frequency per se does not prove effectiveness. Human interactions are very complicated and shaped by multiple factors, including the standard interaction patterns of the surrounding culture and the degree and nature of the rapport between a particular tutor and a particular tutee. Hence, the maximally effective tutoring moves – the ones most worthwhile to mimic in an artificial tutoring system –

might be few and embedded within a lengthy interaction that is rich, varied and complicated for a variety of reasons that bear little causal relation to the learning outcomes.

Past research implicitly assumed that tutoring expertise is general across recipients; that is, if a tutor is effective with student X, he/she tends to be effective with student Y as well. Therefore, we can uncover the moves of effective tutoring by statistical aggregation of code-and-count data over students and sessions but within tutor. However, the richness of human dialogues once again intervenes. It is highly likely that each tutor succeeds better with some students than with others; better rapport, more closely related and matching linguistic habits or thoughts, and so forth. It is therefore reasonable to focus on tutoring sessions, not tutoring persons. This point was strongly brought home to us when we did not find differences between the expert and the novice tutor in the CS domain, even if we had found differences between tutors in the letter pattern domain; rather, we did find great variation in outcomes between different tutoring *sessions*.

The question then becomes, what happens in those sessions in which much learning happened, and what differentiates them from those session in which it didn't? This is clearly a different question than asking what certain persons, expert tutors, tend to do that other persons do not do, or do less of. The next methodological innovation, which develops a trend that is already present in some recent tutoring studies [VanLehn *et al.*, 2003], is to move away from ANOVAs and chi-squares to a correlational approach. To answer the question which categories, e.g. tutor moves, are causally related to the learning outcomes, we employ multiple regression, with the amount of learning per session as the predicted variable and the frequencies of the tutoring behaviors as the predictor variables. The beta weights, the partial regression coefficients, provide information regarding the relative strength of the relations between the relevant tutoring moves and the learning outcomes. This method still relies on the frequency of each type of tutoring move as an important variable, but it does not assume that the more effective tutoring moves are necessarily more frequent, in absolute numbers, than the less effective ones. The method reveals whether *variation* in the frequency of one type of tutoring move is more strongly related to *variation* in learning outcomes than another type of move, regardless of which type of move is more or less frequent. An additional advantage of the correlational methodology is that it will tell us if we are on the wrong track: One possible outcome is that none of the partial regression coefficients are significant. This is a sign that the categories of tutoring moves are not the right ones, and that the transcripts need to be re-coded with a different set of categories (i.e., that the theory behind the category system is wrong and needs to be replaced by a different learning theory). There is no counterpart to this in the code-and-count methodology: Any set of codes will always generate some frequencies, and there will always be some code that turns out to be more frequent than another. There is no built-in warning signal that the category system is fundamentally flawed, but in the correlational approach, low and non-significant correlations do provide such a warning.

This is the approach we are taking in our current award, and we believe it will finally allow us to shed some definite light on which specific features of tutoring dialogues correlate with learning.

References

[Anderson *et al.*, 1995] John R. Anderson, Albert T. Corbett, Kenneth R. Koedinger, and R. Pelletier. Cognitive tutors: Lessons learned. *Journal of the Learning Sciences*, 4(2):167–207, 1995.

[Carletta, 1996] Jean Carletta. Assessing agreement on classification tasks: the Kappa statistic. *Computational Linguistics*, 22(2):249–254, 1996. Squib.

[Chi *et al.*, 2001] Michelene T. H. Chi, Stephanie A. Siler, Takashi Yamauchi, and Robert G. Hausmann. Learning from human tutoring. *Cognitive Science*, 25:471–533, 2001.

[Cohen, 1960] Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20:37–46, 1960.

[Di Eugenio and Glass, 2004] Barbara Di Eugenio and Michael Glass. The Kappa statistic: a second look. *Computational Linguistics*, 30(1):95–101, 2004. Squib.

[Di Eugenio and Trolio, 2005] Barbara Di Eugenio and Michael J. Trolio. Generating driving directions for intelligent vehicles interfaces. In *ECBS05, the 12th Annual IEEE International Conference on the Engineering of Computer Based Systems*, pages 415–422, Greenbelt, MD, April 2005.

[Di Eugenio *et al.*, 2002] Barbara Di Eugenio, Michael Glass, and Michael J. Trolio. The DIAG experiments: Natural Language Generation for Intelligent Tutoring Systems. In *INLG02, The Third International Natural Language Generation Conference*, pages 120–127, Harriman, NY, July 2002.

[Di Eugenio *et al.*, 2005a] Barbara Di Eugenio, Davide Fossati, Dan Yu, Susan Haller, and Michael Glass. Aggregation improves learning: experiments in Natural Language Generation for Intelligent Tutoring Systems. In *ACL05, Proceedings of the 42nd Meeting of the Association for Computational Linguistics*, pages 50–57, 2005.

[Di Eugenio *et al.*, 2005b] Barbara Di Eugenio, Davide Fossati, Dan Yu, Susan Haller, and Michael Glass. Natural Language Generation for Intelligent Tutoring Systems: a case study. In *AIED 2005, the 12th International Conference on Artificial Intelligence in Education*, Amsterdam, The Netherlands, 2005.

[Di Eugenio *et al.*, 2006] Barbara Di Eugenio, Trina C. Kershaw, Xin Lu, Andrew Corrigan-Halpern, and Stellan Ohlsson. Toward a computational model of expert tutoring: a first report. In *FLAIRS06, the 19th International Florida AI Research Symposium*, Melbourne Beach, FL, 2006.

[Di Eugenio, 2001] Barbara Di Eugenio. Natural language processing for computer-supported instruction. *Intelligence*, 12(4), Winter 2001.

[Evens and Michael, 2006] Martha Evens and Joel Michael. *One-on-one Tutoring by Humans and Machines*. Mahwah, NJ: Lawrence Erlbaum Associates, 2006.

[Evens *et al.*, 1993] Martha W. Evens, John Spitkovsky, Patrick Boyle, Joel A. Michael, and Allen A. Rovick. Synthesizing tutorial dialogues. In *Proceedings of the Fifteenth Annual Conference of the Cognitive Science Society*, pages 137–140, Hillsdale, New Jersey, 1993. Lawrence Erlbaum Associates.

[Fox, 1993] Barbara A. Fox. *The Human Tutorial Dialogue Project: Issues in the design of instructional systems*. Lawrence Erlbaum Associates, Hillsdale, NJ, 1993.

[Glass and Di Eugenio, 2002] Michael Glass and Barbara Di Eugenio. MUP: The UIC standoff markup tool. In *The Third SigDIAL Workshop on Discourse and Dialogue*, Philadelphia, PA, July 2002.

[Glass *et al.*, 1999] Michael Glass, Jung Hee Kim, Martha W. Evens, Joel A. Michael, and Allen A. Rovick. Novice vs. expert tutors: A comparison of style. In *MAICS-99, Proceedings of the Tenth Midwest AI and Cognitive Science Conference*, pages 43–49, Bloomington, IN, 1999.

[Glass *et al.*, 2002] Michael Glass, Heena Raval, Barbara Di Eugenio, and Maarika Traat. The DIAG-NLP dialogues: coding manual. Technical Report UIC-CS 02-03, University of Illinois - Chicago, 2002.

[Graesser *et al.*, 1995] Arthur C. Graesser, Natalie K. Person, and Joseph P. Magliano. Collaborative dialogue patterns in naturalistic one-to-one tutoring. *Applied Cognitive Psychology*, 9:495–522, 1995.

[Graesser *et al.*, 2005] A.C. Graesser, N. Person, Z. Lu, M.G. Jeon, and B. McDaniel. Learning while holding a conversation with a computer. In L. PytlikZillig, M. Bodvarsson, and R. Brunin, editors, *Technology-based education: Bringing researchers and practitioners together*. Information Age Publishing, 2005.

[Koedinger *et al.*, 2003] Kenneth R. Koedinger, Vincent Aleven, and Neil T. Heffernan. Toward a rapid development environment for cognitive tutors. In *12th Annual Conference on Behavior Representation in Modeling and Simulation*, 2003.

[Kotovsky and Simon, 1973] K. Kotovsky and H. Simon. Empirical tests of a theory of human acquisition of information-processing analysis. *British Journal of Psychology*, 61:243–257, 1973.

[Krippendorff, 1980] Klaus Krippendorff. *Content Analysis: an Introduction to its Methodology*. Sage Publications, Beverly Hills, CA, 1980.

[Landauer and Dumais, 1997] Thomas K. Landauer and S.T. Dumais. A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review*, 104:211–240, 1997.

[Lepper *et al.*, 1997] M. R. Lepper, M. F. Drake, and T. O'Donnell-Johnson. Scaffolding techniques of expert human tutors. In K. Hogan and M. Pressley, editors, *Scaffolding student learning: Instructional approaches and issues*. Cambridge, MA: Brookline, 1997.

[Litman *et al.*, 2004] Diane J. Litman, Carolyn P. Rosé, Kate Forbes-Riley, Kurt VanLehn, Dumisizwe Bhembe, and Scott Silliman. Spoken versus typed human and computer dialogue tutoring. In *Proceedings of the Seventh International Conference on Intelligent Tutoring Systems*, Maceio, Brazil, 2004.

[Litman *et al.*, 2006] Diane J. Litman, Carolyn P. Rosé, Kate Forbes-Riley, Kurt VanLehn, Dumisizwe Bhembe, and Scott Silliman. Spoken versus typed human and computer dialogue tutoring. *International Journal of Artificial Intelligence in Education*, 16:145–170, 2006.

[Liu *et al.*, 1998] Bing Liu, Wynne Hsu, and Yiming Ma. Integrating classification and association rule mining. In *Knowledge Discovery and Data Mining*, pages 80–86, New York, August 1998.

[Lu, 2006] Xin Lu. Expert tutoring and natural language feedback in intelligent tutoring systems. In *Doctoral Student Consortium at the 14th International Conference on Computers in Education (ICCE2006)*, Beijing, China, December 2006.

[Lu, 2007] Xin Lu. *Expert Tutoring and Natural Language Feedback in Intelligent Tutoring Systems*. PhD thesis, University of Illinois - Chicago, June 2007.

[Mitrović and Ohlsson, 1999] Antonija Mitrović and Stellan Ohlsson. Evaluation of a constraint-based tutor for a data-base language. *International Journal of Artificial Intelligence and Education*, 10:238–256, 1999.

[Mitrović and Ohlsson, 2006] Antonija Mitrović and Stellan Ohlsson. A critique of kodaganallur, weitz and rosenthal, "a comparison of model-tracing and constraint-based intelligent tutoring paradigms". *International Journal of Artificial Intelligence in Education*, 16(3):277–289, 2006.

[Mitrović *et al.*, 2006] Antonija Mitrović, Stellan Ohlsson, and Brent Martin. Problem-solving support in constraint-based tutors. *Technology, Instruction, Cognition and Learning*, 3(1), 2006.

[Moore *et al.*, 1996] Johanna D. Moore, Benoît Lemaire, and James A. Rosenbloom. Discourse generation for instructional applications: Identifying and exploiting relevant prior explanations. *Journal of the Learning Sciences*, 5(1):49–94, 1996.

[Nokes and Ohlsson, 2005] Timothy J. Nokes and Stellan Ohlsson. Comparing multiple paths to mastery: What is learned? *Cognitive Science*, 29:769–796, 2005.

[Ohlsson, 1992] Stellan Ohlsson. Constraint-based student modeling. *Journal of Artificial Intelligence and Education*, 3(4):429–447, 1992. Reprinted in J. E. Greer and G. I. McCalla (Eds.), Student modeling: The key to individualized knowledge-based instruction, Springer Verlag, 1994.

[Ohlsson, 2007] Stellan Ohlsson. *Deep Learning: How the Mind Overrides Past Experience*. Cambridge University Press, 2007. (In preparation).

[Person, 2006] Natalie Person. Why study expert tutors? Presentation at ONR Contractors' Conference on Instructional Strategies, February 2006.

[Peters *et al.*, 2004] Stanley Peters, Elizabeth Owen Bratt, Brady Clark, Heather Pon-Barry, and Karl Schultz. Intelligent systems for training damage control assistants. In *Proceedings of I/ITSEC 2004, Interservice/Industry Training, Simulation, and Education Conference*, Orlando, Florida., 2004.

[Poesio *et al.*, 2006] Massimo Poesio, Barbara Di Eugenio, and Amrita Patel. Discourse structure and anaphora in tutorial dialogues: an empirical analysis of two theories of the global focus. *Research on Language and Computation*, 4(2–3):229–257, 2006.

[Pon-Barry *et al.*, 2006] Heather Pon-Barry, Karl Schultz, Elizabeth Owen Bratt, and Stanley Peters. Responding to student uncertainty in spoken tutorial dialogue systems. *International Journal of Artificial Intelligence in Education*, 16:171–194, 2006.

[Rosé *et al.*, 2003] C. P. Rosé, D. Bhembe, S. Siler, R. Srivastava, and K. VanLehn. Exploring the effectiveness of knowledge construction dialogues. In *AIED03, Proceedings of AI in Education*, 2003.

[Scott, 1955] William A. Scott. Reliability of content analysis: The case of nominal scale coding. *Public Opinion Quarterly*, 19:127–141, 1955.

[Serafin and Di Eugenio, 2004] Riccardo Serafin and Barbara Di Eugenio. FLSA: Extending Latent Semantic Analysis with features for dialogue act classification. In *ACL-EACL04, 42nd Meeting of the Association for Computational Linguistics*, Barcelona, Spain, 2004.

[Thompson and McKelvie, 1997] Henry Thompson and David McKelvie. Hyperlink semantics for standoff markup of read-only documents. In *SGML Europe 97, Barcelona*, 1997.

[Towne, 1997] Douglas M. Towne. Approximate reasoning techniques for intelligent diagnostic instruction. *International Journal of Artificial Intelligence in Education*, 1997.

[VanLehn *et al.*, 2003] Kurt VanLehn, Stephanie Siler, and Chaz Murray. Why do only some events cause learning during human tutoring? *Cognition and Instruction*, 21(3):209–249, 2003.

[Zakharov *et al.*, 2005] K. Zakharov, S. Ohlsson, and A. Mitrović. Feedback Micro-Engineering in EER-Tutor. In C-K Looi, G. McCalla, B. Bredeweg, and J. Breuker, editors, *AIED 2005, Proc. 12th Conference on Artificial Intelligence in Education*, pages 718–725, 2005.

[Zinn *et al.*, 2005] Claus Zinn, Johanna D. Moore, and Mark G. Core. Intelligent information presentation for tutoring systems. In Massimo Zancanaro and Oliviero Stock, editors, *Multimodal Intelligent Information Presentation*, pages 227–254. Dordrecht: Kluwer Academic Publishers, 2005.